(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) **EP 1 063 830 B1**

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
07.03.2007 Bulletin 2007/10

(51) Int Cl.:
H04L 29/06 *(2006.01)*

(21) Application number: **00305250.3**

(22) Date of filing: **21.06.2000**

(54) **Method and apparatus for multiplexing payload data in a data network**

Verfahren und Vorrichtung zur Multiplexierung der Nutzlastdaten in einem Datennetzwerk

Procédé et appareil pour le multiplexage de données de charge utile dans un réseau de données

(84) Designated Contracting States:
DE FR GB

(30) Priority: **23.06.1999 US 338531**

(43) Date of publication of application:
27.12.2000 Bulletin 2000/52

(73) Proprietor: **Nortel Networks Limited**
St.Laurent,
Québec H4S 2A9 (CA)

(72) Inventors:
- **Luo, Gang**
  Kanata,
  Ontario K2M 2N4 (CA)
- **Giese, Peter A.**
  Kinburn,
  Ontario K0A 2H0 (CA)
- **Davidson, Paul G.**
  Ashton
  Ontario KOA 1BO (CA)

(74) Representative: **Ertl, Nicholas Justin et al**
Elkington and Fife,
Prospect House,
8 Pembroke Road
Sevenoaks,
Kent TN13 1XR (GB)

(56) References cited:
- **HANDLEY M: "GeRM: Generic RTP Multiplexing" INTERNET DRAFT, [Online] 11 November 1998 (1998-11-11), XP002139359 Retrieved from the Internet: <URL:ftp://ftp.informatik.uni-bremen.de/pu b/doc/internet-drafts/draft-ietf-avt-germ- 00.ps> [retrieved on 2000-09-29]**
- **CAMERON P ET AL: "Transport Multiplexing Protocol (TMux)" REQUEST FOR COMMENTS 1692, [Online] August 1994 (1994-08), XP002149445 Retrieved from the Internet: < URL:ftp://ftp.isi.edu/in-notes/rfc1692.tx t> [retrieved on 2000-09-29]**
- **ROSENBERG J; SCHULZRINNE H: "An RTP Payload Format for User Multiplexing" INTERNET DRAFT, [Online] 6 May 1998 (1998-05-06), XP002127739 Retrieved from the Internet: < URL:http://www.cs.columbia.edu/~jdrosen/p apers/draft-ietf-avt-aggregation-00.txt> [retrieved on 2000-10-04]**
- **HOSHI T ET AL: "VOICE STREAM MULTIPLEXING BETWEEN IP TELEPHONY GATEWAYS" IEICE TRANSACTIONS ON INFORMATION AND SYSTEMS,JP,INSTITUTE OF ELECTRONICS INFORMATION AND COMM. ENG. TOKYO, vol. E82-D, no. 4, April 1999 (1999-04), pages 838-845, XP000832566 ISSN: 0916-8532**
- **SUBBIAH B; SENGODAN S: " User Multiplexing in RTP payload between IP Telephony Gateways" INTERNET DRAFT, [Online] 21 August 1998 (1998-08-21), XP002127741 Retrieved from the Internet: <URL:http://www.ietf.org/ proceedings/99mar /I-D/draft-ietf-avt-mux-rtp-00.txt> [retrieved on 2000-09-29]**
- **ROSENBERG J; SCHULZRINNE H: "Issues and Options for RTP Multiplexing" INTERNET DRAFT, [Online] 1 October 1998 (1998-10-01), XP002149446 Retrieved from the Internet: < URL:ftp://ftp.informatik.uni-bremen.de/pu b/doc/ internet-drafts/draft-ietf-avt-muxis sues-00.txt> [retrieved on 2000-09-29]**

Note: Within nine months from the publication of the mention of the grant of the European patent, any person may give notice to the European Patent Office of opposition to the European patent granted. Notice of opposition shall be filed in a written reasoned statement. It shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European Patent Convention).

Printed by Jouve, 75001 PARIS (FR)

(Cont. next page)

- **EL-KHATIB K; LUO G; BOCHMANN G: "Multiplexing Scheme for RTP Flows between Access Routers" INTERNET DRAFT, [Online] 24 June 1999 (1999-06-24), XP002149447 Retrieved from the Internet: <URL:ftp://ftp.informatik.uni-bremen.de/pu b/doc/internet-drafts/draft-ietf-avt-multi plexing-rtp-00.txt> [retrieved on 2000-09-29]**
- **CASNER S ET AL: "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links" INTERNET DRAFT, [Online] 27 July 1998 (1998-07-27), XP002125101 Retrieved from the Internet: < URL:ftp://ftp.kyoto.wide.ad.jp/multicast/ docs/ drafts/draft-ietf-avt-crtp-05.txt> [retrieved on 2000-09-29]**

Description

[0001] The present invention relates to data networks, and more particularly to data network signals, methods and devices that are suitable for multiplexing payload data in a packet switched data network.

[0002] In recent years, packet switched data networks have become heavily used and relied upon. The best known of such networks use the known Internet protocol ("IP") as detailed in the Internet Engineering Task Force, Request for Comment ("RFC") 791, the contents of which are hereby incorporated by reference. Numerous protocols based on IP are used to carry a variety of payload data on IP compliant networks. These include the Uniform Datagram Protocol ("UDP") and the Real-Time Protocol ("RTP") as detailed in RFCs 768 and 1889, the contents of which are hereby incorporated by reference.

[0003] All of these protocols utilize headers to identify packets and attributes of packets transported across the network. Such headers introduce overhead. For example, common IP telephony techniques as detailed in International Telephony Union ("ITU") Recommendation H.323, the contents of which are hereby incorporated by reference, rely on the RTP to transport payload voice data. RTP in turn relies on the UDP and the IP. Each RTP/UDP/IP compliant voice packet typically includes a total of forty (40) bytes of overhead in the header.

[0004] By contrast, modern voice compression techniques compress voice payload to lower and lower bit rates. ITU Recommendation G.723.1, the contents of which are hereby incorporated by reference, for example, compresses voice data to 5.3 kbps. This results in voice data that is typically packetized every thirty (30) ms, with each packet having a payload of twenty (20) bytes. Thus, using the RTP may result in payload occupying only one-third of each packet, with the remaining two-thirds of the packet dedicated to protocol overhead. Other packet based protocols similarly often use only ten (10) to twenty (20) bytes of data.

[0005] It has been recognized that multiple voice streams are typically exchanged concurrently between two IP telephony gateways. Thus, voice payload for multiple streams may be combined or multiplexed within packets reducing the overhead to payload data ratio for each packet. Proposals based on such multiplexing include K. Tanigawa, T. Hoshi and K. Tsukada: A RTP simple multiplexing transfer method for Internet telephony gateway, IETF draft, work in progress June 1998; J. Rosenberg and H. Schulzrinne: An RTP payload format for user multiplexing, IETF draft, work in progress, Aug. 21, 1998; and Mark Handley (ISI), AVT group meeting minutes for Aug. 1998 meeting.

[0006] These proposals suggest stripping existing RTP/UDP/IP headers at network gateways, mapping these to mini-headers included in multiplexed packets, and transferring the multiplexed packets including mini-headers in a single IP packet to a recipient gateway. In advance of transferring multiplexed packets, the relationship between RTP/UDP/IP headers and mini-headers is communicated between gateways, typically using out-of-band signalling. The recipient gateway replaces each received mini-header with an associated full RTP/UDP/IP header and passes the reconstructed RTP packets to computing devices on the far side of the recipient gateway.

[0007] These approaches rely on the presence of gateways, such as those defined in ITU Recommendation H.323. Moreover, they typically require modifications to control protocols to exchange mapping information between gateways. However, many IP telephony applications and similar low bit rate applications are end-to-end, and do not rely on gateways.

[0008] Accordingly, an improved method of multiplexing data within packets, and a protocol making use of such a method are desirable.

[0009] The article "GeRM: Generic RTP Multiplexing", internet Draft of the Audio Video Trasnport Working Group of the Internet Engineering Taskforce, of 18 November 1998, and by Mark Handley (XP002139359) discloses a payload format for the multiplexing of multiple RTP streams. The multiplexed packet encodes only differences between the RTP headers of the different payloads in the packet.

[0010] The invention provides a method of forming a mapping table as defined by claim 1, a computing apparatus storing a mapping table as defined by claim 11, a computer readable memory as defined by claim 17 or 21 and a method of multiplexing payload data in a plurality of packets as defined by claim 20.

[0011] In accordance with the present invention, payload associated with multiple packets is multiplexed into a single multiplexed packet. Each payload portion is identified by a mini-header within the multiplexed packet. Advantageously, mapping information is also be transferred as part of such multiplexed packets which include multiplexed payload data. Preferably, the mapping information is used to form mapping tables within routers at edges of access networks. The mapping tables may be used to establish a relationship between mini-headers and full headers. The mapping tables may be used to multiplex data from packets to form a multiplexed packet at an ingress router, and demultiplex the multiplexed packet at an egress router. Conveniently, neither gateways nor out of band signalling are required.

[0012] Payload data contained in a plurality of packets to be passed by way of a network node within a packet switched data network, can be multiplexed by associating with the payload portion of each packet a mini-header smaller than a header for the packet. A multiplexed packet including each payload portion and an associated mini-header, and further including a mapping token, which establishes a relationship between a mini-header and a header for one of the packets, for which this relationship is not known at the network node is constructed.

[0013] The mapping table may be formed at a network node within the packet switched network. The mapping

table maps mini-headers identifying multiplexed payload data from several packets within a multiplexed packet, to complete headers usable to transport payload on the network. An entry of the mapping table may be formed by receiving a packet including at least one payload data portion from one of the packets; a mini-header associated with the payload data portion; and a mapping token, indicative of a relationship between the mini-header and a complete header for transporting the data portion on the network. The formed entry within the table, includes a portion of the mini-header and information derived from the mapping token, indicative of the a relationship between the mini-header and the complete header.

[0014] The computing apparatus of the invention is preferably interconnected with a packet switched data network, and preferably includes computer readable memory that stores a mapping table for mapping mini-headers identifying multiplexed payload data from several packets within a multiplexed packet to a complete header usable within the packet switched data network. As well, the computer readable memory stores computer readable instructions, that adapt the apparatus to process multiplexed packets including, payload portions; mini-headers associated with each payload portion, usable in conjunction with the mapping table to construct a packet including the associated payload that may be transported on the packet switched data network; and at least one mapping token, usable by the apparatus to update the mapping table. Preferably the computing apparatus includes a router on the network.

[0015] The computer readable memory of the invention preferably stores computer executable instructions, that may be executed by a computing device interconnected with a packet switched network. The computer readable instructions adapt a computing device to maintain a mapping table for mapping mini-headers identifying multiplexed payload data from several packets within a multiplexed packet to a complete header usable within the network. Moreover, they adapt the computing device to process a multiplexed packet which includes a payload portion, formed from a packet transported on the network; a mapping header associated with the payload portion within the multiplexed packet and usable by the computing device in conjunction with the mapping table to construct a packet including the associated payload that may be transported on the network; and mapping information, usable by the device to update the mapping table.

[0016] In the multiplexing method of the invention, payload data contained in a plurality of packets to be passed by way of a network node in a switched packet network, is multiplexed, by searching a table with information in a header of each packet for a match. If a match for a given packet is not found, information from its header is entered in the table in association with information for a mini-header which comprises a channel identifier, identifying payload data intended to be exchanged between the same source and the same destination, and a length field indicating the length of said associated payload data portion. A mapping token associating the given header information with the mini-header is also created, the mapping token including a channel identifier, a payload length field indicating a length of zero and the header of one of said packets. The payload portion of the packet is extracted and the header is discarded. The mini-header is associated with the payload portion, and a multiplexed packet including the mapping token is constructed.

[0017] Other aspects and features of the present invention will become apparent to those of ordinary skill in the art, upon review of the following description of specific embodiments of the invention in conjunction with the accompanying figures.

[0018] Exemplary embodiments of the present invention will now be described with reference to the accompanying drawings, in which:

FIG. 1 illustrates a computing network, including routers, that can support the various underlying inventive concepts of the preferred embodiment of the present invention;
FIG. 2 is a block diagram of an exemplary architecture of routers of FIG.1;
FIG. 3 is a block diagram of an exemplary memory organization of routers of FIG. 1;
FIG. 4 illustrates an example ingress mapping table formed at a router of FIG. 1;
FIG. 5 illustrates an example egress mapping table formed at a router of FIG. 1;
FIG. 6 illustrates an exemplary format of a multiplexed packet;
FIG. 7 illustrates an example format of a mini-header used in the packet of FIG. 6;
FIG. 8 illustrates an example format of a mapping establishment token used in the packet of FIG. 6;
FIG. 9 is a flow chart of steps performed at an ingress router; and
FIGS. 10 and 11 illustrate steps performed at an egress router.

[0019] FIG. 1 illustrates a packet switched data network 10, including a backbone network 12, in communication with access networks "A" and "B" 18 and 20, by way of access routers "A" and "B" 14 and 16, exemplary of embodiments of the present invention. Example computing devices A1 and A2 22 and 24 are in communication with access network "A" 18; similarly example computing devices B1 and B2 26 and 28 are in communication with access network "B" 20.

[0020] Computing devices 22, 24, 26, 28 may be general purpose computing devices including a processor, memory and network interface (all not illustrated). The memory may store software including operating system software and application software, which in turn may include an IP stack. The application software may include ITU Recommendation H.323 compliant client software, allowing computing devices 22, 24, 26 and 28 to establish end-to-end voice over IP telephony sessions with each

other.

[0021] Backbone network 12 is preferably an IP compliant data network in accordance with RFC 791, including conventional packet routers 30 and 32. Backbone network 12 could, for example, be the public Internet. Access networks "A" and "B" 18 and 20 are similarly IP compliant data networks. These may be corporate intranets, or simply portions of the public Internet. Access routers "A" and "B" 14 and 16 interconnect access networks "A" and "B" 18 and 20 with backbone network 12, respectively. As illustrated, access routers 14 and 16 are in direct communication with routers 30 and 32 of backbone network 12. Routers 14, 16, 30 and 32 are thus nodes within network 10. Access routers "A" and "B" 14 and 16 may, for example, be conventional Nortel Networks Passport switches/routers adapted to function in manners exemplary of the present invention.

[0022] As will become apparent, access routers "A" and "B" 14 and 16 switch packets received from the remainder of access network "A" and "B" to backbone network 12. Similarly, access routers 14 and 16 switch packets received from backbone network 12 to access networks "A" and "B". As well, access routers "A" and "B" 14 and 16 multiplex, and optionally demultiplex, suitable packets in manners exemplary of the present invention.

[0023] An example hardware architecture of access router "A" 14 is illustrated in block diagram in FIG 2. The architecture of routers 16, 30 and 32 may be substantially similar to that of access router "A" 14. As illustrated in FIGS. 1 and 2, access router "A" may include a processor 34, in communication with storage memory 36 and a plurality of input/output ("I/O") ports 38. I/O ports 38 interconnect router 14 with the remainder of access network "A" and with routers of backbone network 12 (FIG. 1). I/O ports may be conventional DS1; OC-3; Ethernet or other suitable interfaces. Storage memory 36 may be any suitable combination of random access memory; disk memory; read-only memory; or any other suitable computing memory known to those of ordinary skill in the art. Processor 34, in turn, may be a conventional microprocessor such as an INTEL x86 processor or a Motorola MC68xxxx processor. Of course any other suitable processor may be used as part of router "A" 14.

[0024] An example organization of memory 36 is illustrated in FIG. 3. As illustrated, memory 36 stores an operating system 40; application software 42; a routing table 44; an ingress mapping table 46; and an egress mapping table 48. Operating system 40 in combination with application software 42 allow access router "A" 14 to route packets in a conventional manner, as well as perform in accordance with methods exemplary of the present invention.

[0025] Routing table 44 is formed by application software 42, and allows router 14 to route packets from one of I/O ports 38 to another based on a packet's destination address. Routing table 44 may, for example, be formed at router "A" 14 using the OSPF routing protocol, as detailed in RFC 2328, the content of which are hereby incorporated by reference, implemented by a portion of application software 42. Ingress mapping table 46 and egress mapping table 48 are also formed by application software 42 in manners exemplary of the present invention, and as detailed below.

[0026] Ingress mapping table 46 establishes a one-to-one mapping of source IP address, destination IP address, and ports to mini-headers, to allow multiplexing of suitable packets originating with access network "A" 18, prior to entering backbone network 12. That is, the payload of any packet of suitable length originating with a particular source IP address within access network "A" destined for a particular destination address is associated with a mini-header. An organization of ingress mapping table 46 is illustrated in FIG. 4. As illustrated, along with each IP source address to destination address (including logical port number), as stored in fields 50 and 52 of ingress mapping table 46, a payload type ("PT") field 54; an ingress router port number ("IRouter Port#") field 56; and egress router ("ERouter") IP address and port number field 58; a channel identifier ("CID") field 60; and a last-time refreshed ("LTR") field 62 are stored within ingress mapping table 46. As should be appreciated, each IP source address to destination address (including logical port number), as stored in fields 50 and 52 may correspond to a single IP telephony connection across network 10. This connection is thus associated with a channel identifier stored in field 60.

[0027] An example organization of egress mapping table 48 is illustrated in FIG. 5. Egress mapping table 48 similarly establishes a one-to-one mapping of mini-headers to source IP address; destination IP address and ports, to demultiplex multiplexed packets received at router "A", exiting from backbone network 12. As illustrated, for each channel ID field 70, egress mapping table 48 stores an ingress router IP address and port number in field 72; an egress router port number in field 74; a full RTP/UDP/IP header in field 76; a payload type in field 78; a last time refreshed in field 80; and a last packet reproduced timestamp and sequence number in field 82.

[0028] Router "B" 16 acting as an egress or ingress router similarly forms egress and ingress mapping tables of the form of ingress an egress mapping tables 46 and 48. Conventional routers 30 and 32 will typically not form similar egress and ingress mapping tables.

[0029] Router "A" 14 acting as an ingress router forms IP packets containing payload data associated with multiple RTP packets, to form multiplexed packets having the format of multiplexed packet 90 as illustrated in FIG. 6. As illustrated, a multiplexed packet 90 includes a conventional UPD/IP header 91 followed by a plurality of payload portions 92a, 92b and 92c, each associated with a mini-header 94a, 94b and 94c. Each mini-header/payload portion may be considered a channel within multiplexed packet 90. Further, some mini-header/payload combinations such as mini-header/payload of fields 92c and 94c may further be associated with a mapping establishment token 96. It will be appreciated that multi-

plexed packet 90 may have significantly more or less than three multiplexed payload fields, as illustrated.

[0030] The format of each mini-header 94a, 94b and 94c (collectively and individual mini-header 94) is illustrated in FIG. 7. As illustrated each mini-header preferably includes an eight bit channel identifier field 98 and a further eight bit payload length field 100. Optionally, each mini-header 94 may additionally include an RTP time stamp and sequence number taken from the original RTP/UDP/IP header associated with the payload data.

[0031] The format of each mapping establishment token is illustrated in FIG. 8. As illustrated, each mapping establishment token includes an channel identifier field 102; a payload length field, indicating a length of zero (0) 104; and a full RTP/UDP/IP header in field 106.

[0032] In operation, a computing device, such as computing device A1 22 on access network "A" 18 illustrated in FIG. 1 wishes to exchange packets each having a relatively small payload with a computing device B1 26 on access network "B" 20. For example, computing device A1 22 may include an Internet telephony application that is compliant with H.323. As such, computing device A1 22 may form packets having a payload of twenty (20) bytes with an associated RTP/UDP/IP header. The RTP/UDP/IP header identifies computing device B1 26 on access network "B" 20, as well as an appropriate logical port of device B1 26. At the same time, computing device A2 24 may wish to exchange similar small payload packets with computing device B2 28.

[0033] Consequently, RTP/UDP/IP compliant packets originating with devices A1 and A2 22 and 24, identifying devices B1 and B2 26 and 28 are forwarded to access router "A" 14, which acts as an ingress router to backbone network 12. The steps S900 performed by access router "A" 14 under control of software 42 may be better understood with reference to FIG. 9. Specifically, access router "A" 14 receives an RTP/UDP/IP packet in step S902 and determines if the received packet has "small" payload by, for example, examining the RTP/UDP/IP header associated with the received packet in step S904. In the preferred embodiment only payload data associated with packets having less than 256 bytes of payload will be multiplexed. However, packets with smaller or larger payload could be multiplexed. If the received packet is above this threshold size, it is passed to backbone network 12 in step S906, without being multiplexed or otherwise modified.

[0034] Next, for a "small" payload packet, access router "A" 14 examines source and destination packet addresses and ports for the RTP/UDP/IP packet, and ingress mapping table 46 stored in memory 36 in step S906. If any of the entries in fields 50 and 52 of ingress mapping table 46 (FIG. 4) correspond to the source and destination addresses and ports of the RTP/UDP/IP header, access router 14 associates the channel number within field 60 to form a mini-header associated with the RTP/UDP/IP payload having the format of mini-header 94 of FIG. 7, in step S910. Length field 100 is filled with the payload length of the RDP/UDP/IP packet derived from this RDP/UDP/IP header. Access router "A" 14 determines an IP address and port of a destination egress router for the payload data from field 58. In the preferred embodiment, multiplexed packets destined for a particular egress router will be destined for a unique IP address and port at the egress router, dedicated to the receipt of multiplexed packets and stored within field 58. Router "A" then forms a multiplexed packet destined for the destination egress router including the formed mini-header/payload combination. Access router "A" 14 also discards the full RTP/UDP/IP header associated with the payload. Preferably, access router "A" 14 will buffer each multiplexed packet for at least five milliseconds, and preferably no more than ten milliseconds. However, it may buffer the multiplexed packet for more or less time. An ideal buffer time may be experimentally determined. Alternatively, if a multiplexed packet destined for the identified egress router is already buffered, router "A" may append the formed mini-header/payload combination to an already buffered packet. If within the buffering interval, additional RTP/UDP/IP packets arrive, such as a packet from computing device A2 24, and a similar mini-header is formed, destined for the same egress router, access router "A" 14 forms a multiplexed packet as illustrated in FIG. 6, containing the first and subsequent mini-headers and payload.

[0035] The multiplexed packet is then forwarded to router 30 of backbone network 12, in a conventional manner, and is then eventually routed to its destination by way of router 32 and access router "B" 16 to an associated logical port. Router "B" uses it egress table 48 including channel ID entries to reconstitute a full header, as explained more fully below.

[0036] If, on the other hand, table 46 of access router "A" acting as an ingress router does not contain an entry corresponding to a received RTP/UDP/IP header received from computing device A1 22, access router "A" 14 initiates a query to an access router, such as access router "B" 16, associated with the RTP/UDP/IP destination address to determine whether the egress router supports a multiplexing protocol in steps S914 and S916. The query may be initiated by way of UDP message to router "B" 16, or in any other way known to those of ordinary skill in the art. As will be appreciated by those of ordinary skill, a destination access router "B" 16 associated with a destination IP address may be identified in step S914 at access router "A" using routing table 44.

[0037] If access router "B" does not respond to the query, or responds negatively as determined in step S918, the RTP/UDP/IP packet is dispatched from access router "A" 14 in a conventional manner, as in step S906, without multiplexing or modification. If access router "B" 16 responds affirmatively, access router "A" creates a new entry in table 46 associated with the source and destination IP address pair and assigns a new channel ID in field 60 to the source/destination pair in step S920. Next, access router "A" 14 forms a mapping establish-

ment token having the format of token 102, in FIG. 8 including the assigned channel ID and RTP/UDP/IP header of the received packet. Thereafter, the mapping establishment token, is included in the multiplexed packet in step S922. Next, access router "A" forms a mini-header for the payload data, as illustrated in FIG. 7 in step S910 and includes the mini-header/payload for the received packet in the multiplexed packet, as in step S910. Preferably, the mapping establishment token precedes the mini-header/payload combination within the multiplexed packet as illustrated in FIG. 6. Within a multiplexed packet 90, a mapping establishment token can be distinguished from a mini-header by way of the contents of length field 104. For a mapping establishment token 96, length field 104 will indicate a length of zero (0). For a mini-header, having the format of mini-header 94, the equivalent field 100 will indicate a non-zero length. Alternatively, the length of the CID field of mini header 94 could be reduced to seven bits, allowing the last bit to be used to identify a mapping token.

[0038] At the end of the buffer interval for the multiplexed packet, the multiplexed packet is dispatched by access router "A" 14 in a conventional manner. As should be apparent, the UDP/IP header 91 of the multiplexed packet identifies access router "A" as its source, and access router "B" as its destination. The multiplexed packet is routed across backbone network 12 to its eventual destination address identifying router "B" 16.

[0039] Conveniently, then, with a proper choice of a buffering interval, payload associated with multiple RTP/UDP/IP packets associated with multiple IP connections between access network "A" and access network "B" will be combined and multiplexed in each multiplexed packet. Moreover, as mapping establishment tokens are contained within the multiplexed packets, no out of band signalling is required. As such, this multiplexing is compatible with existing protocols, such as those defined by ITU Recommendation H.323, without any required modification.

[0040] Steps S1000 and S1100 performed by access router "B" 16 upon receipt of a packet, including a multiplexed packet originating with access router "A" 14, may be better understood with reference to FIGS. 10 and 11. That is, router "B" upon receipt of any packet in step S1002 determines if it is directed to the known logical port and IP address reserved for multiplexed packets in step S1004. If not, the packet is processed in a conventional manner in step S1006. That is, the packet may be passed to a further network interconnected device without additional processing at access router "B" 16.

[0041] If the packet identifies the IP address and port reserved for multiplexed packets, router "B" demultiplexes the received packet in accordance with steps S1100 illustrated in FIG. 11. That is, router "B" 16 sequentially examines the contents of the received packet having the format of packet 90 to locate a mini-header 94 or a mapping establishment token 96. If router "B" initially locates a mapping establishment token in step S1102, it uses the contents of this header to update the contents of its egress mapping table, having the format of egress mapping table 48, as illustrated in FIG. 5. That is, as it encounters mapping establishment token, it creates a new entry within its egress mapping table including the RTP/UDP/IP header. The channel ID field 102 of the mapping establishment token may be used to complete field 70 of this entry. The contents of the RTP/UDP/IP header may be used to complete the payload type field 78; ingress router IP address and port field 72; and RTP/UDP/IP header field 76 of the entry. As will be appreciated, the next RTP/UDP/IP packet associated with the recently stored mini-header may now be identified using the channel ID now stored at access routers "A" and "B" 14 and 16.

[0042] Once a mini-header is encountered within the received packet, access router "B" 16 under control of application software 42 extracts the mini-header in step S1106 to associate a full RTP/UDP/IP header, as stored in field 76 of egress mapping table 48 with each payload portion of the multiplexed packet, to reconstruct a full RTP/UDP/IP packet.

[0043] If the mini-header includes a timestamp and sequence number, these are used to reconstruct the RTP/UDP/IP sequence number and timestamp. Alternatively, the contents of field 82 may be used to reconstruct suitable sequence numbers and timestamps. Specifically, upon creation of an entry reflective of a particular source and destination IP address pair, field 82 may be filled using the sequence number and time stamps of the received RTP/UDP/IP header. Thereafter, field 82 may be incremented appropriately each time a mini header associated with the same channel ID is received. The timestamp may be incremented using the local timer of router "B" 16.

[0044] The resulting reconstructed RTP/UDP/IP packet is then forwarded by router "B" 16 to a proper destination computing device, such as devices B1 and B2 26 and 28 interconnected with access network "B", in step S1112. Steps S1102 and onward are repeated until the entire packet is processed.

[0045] As mapping establishment tokens are included in packets in advance of mini-headers defined by the tokens, mini-headers for which an entry does not exist within egress mapping 48 should not be encountered. However, if a packet including a mini-header for which an entry does not exist is encountered, the mini-header and associated data is discarded. Moreover, full RTP/UDP/IP headers need not be sent once suitable mapping table entries are established.

[0046] Conveniently, access routers "A" and "B" 14 and 16 may also update the last time refreshed ("LTR") field 80,62 of egress mapping table 48, and ingress mapping table 46 any time a defined channel identifier is reused. As well, periodically, at pre-determined intervals routers "A" and "B" may "flush" ingress and egress tables, deleting any entries that have not been used for a defined period, as determined with reference to the last time re-

fresh field 62 and 80. This way, assigned channel identification numbers may be reused. This is particularly convenient as only eight bits are assigned to the channel identification header. Unique channel identification numbers may thus be re-used over time. In the context of IP telephony, and with an appropriate choice of refresh intervals, ingress and egress mapping tables may thus be representative of established and active calls.

[0047] While in the above described embodiment, multiplexing and demultiplexing are performed at access routers "A" 14 and access router "B" 16, a person skilled in the art will appreciate that such multiplexing and demultiplexing, and maintenance of mapping tables could easily be performed at other network nodes, preferably at the "edge" of access networks "A" and "B". For example, routers 30 and 32 could be used to multiplex and demultiplex multiplexed packets. Moreover, while the invention is particularly well suited for use with IP telephony data, it will be appreciated that the invention may be used in association with other types of data, and will typically provide benefits when overhead to payload ratio is large. The network with which it may be used, need not conform to the IP, but may conform to successors of this protocol or some other packet switched protocol.

[0048] While the organization of software blocks, steps, payload data and header structures have been illustrated as clearly delineated, a person skilled in the art will appreciate that the delineation between blocks and structures is somewhat arbitrary. For example, software used to route, multiplex and demultiplex could be formed in hardware. Numerous other arrangements of software blocks and structures are possible.

[0049] Finally, it will be understood that the described embodiments are intended to be illustrative, and in no way limiting. The embodiment are susceptible to modification of form, arrangement of parts, steps, details and order of operation.

Claims

1. A method of forming a mapping table (46,48) at a network node within a packet switched network, said mapping table for mapping mini-headers identifying multiplexed payload data from several packets within a multiplexed packet, to complete headers usable to transport payload on said network, said method comprising:

receiving a packet (90) comprising:

at least one payload data portion (92a,b,c) from one of said packets;
a mini-header (94a,b,c) comprising a channel identifier (98), identifying payload data intended to be exchanged between the same source and the same destination, and a length field (100) indicating the length of said associated payload data portion;
a mapping token (96) including a channel identifier (102), a payload length field indicating a length of zero (104) and the header of one of said packets (106), the mapping token being indicative of a relationship between said mini-header (94a,b,c) and a complete header (91) for transporting said at least one data portion on said network;

forming a new entry within said table (46,48), including a portion (70) of said mini-header and information (76) derived from said mapping token, indicative of a relationship between said mini-header and said complete header (91).

2. The method of claim 1, wherein said mini-header comprises a channel identifier (98,60,70), identifying a logical channel within said multiplexed packet.

3. The method of claim 2, wherein said channel identifier (98,60,70) is stored within said mapping table (46,48).

4. The method of claim 1, wherein said network node comprises a router on said packet switched data network.

5. The method of claim 1 wherein said mapping table is updated each time a mini-header (94a,b,c) including an already defined channel identifier(98,60,70) is received, indicating network communication associated with said logical channel is active.

6. The method of claim 5, further comprising deleting an entry for which a mini-header (94a,b,c) including an already defined channel identifier is not received for a defined interval.

7. The method of claim 1, wherein the channel identifier is 8 bits in length.

8. The method of any preceding claim, wherein the payload data portion (92a,b,c) for each multiplexed packet is less than or equal to 256 bytes in length.

9. The method of any preceding claim, wherein said mini-header (94a,b,c) is shorter than the complete header (91).

10. the method of claim 9, wherein the mini-header (94a, b,c) is 16 bits in length.

11. A computing apparatus interconnected with a packet switched data network, comprising computer readable memory storing
a mapping table (46,48) for mapping mini-headers identifying multiplexed payload data from several

packets within a multiplexed packet to a complete header (91) usable within said packet switched data network;
computer readable instructions, adapting said apparatus to process multiplexed packets comprising:

one or more payload portions (92a,b,c), formed from packets on said network;
one or more mini-headers (94a,b,c) comprising a channel identifier (98), identifying payload data intended to be exchanged between the same source and the same destination, and a length field (100) indicating the length of said associated payload data portion, the mini-headers being usable by said apparatus in conjunction with said mapping table (46,48) to construct a packet including said associated payload that may be transported on said packet switched data network;
at least one mapping token (96) including a channel identifier (102), a payload length field indicating a length of zero (104) and the header of one of said packets (106), the mapping tokens being usable by said apparatus to update said mapping table, the mapping token (96) being indicative of a relationship between said mini-header (94a,b,c) and the complete header (91) for transporting said payload portion on said network.

12. The apparatus of claim 11, wherein said computer readable instructions adapt said apparatus to demultiplex multiplexed packets, using said mini-headers (94a,b,c), said payload portions (92a,b,c) and said mapping tables (46,48) to construct packets transportable on said network and containing said payload data.

13. The apparatus of claim 11, wherein the channel identifier is 8 bits in length.

14. The apparatus of any one of claims 11 to 13, wherein the payload data portion (92a,b,c) for each multiplexed packet is less than or equal to 256 bytes in length.

15. The apparatus of any one of claims 11 to 14, wherein said mini-header (94a,b,c) is shorter than the complete header (91).

16. The apparatus of claim 15, wherein the mini-header (94a,b,c) is 16 bits in length.

17. A computer readable memory storing computer executable instructions, executable by a computing device interconnected with a packet switched network, said computer readable instructions, adapting said computing device to:

maintain a mapping table (46,48) for mapping mini-headers (94a,b,c) identifying multiplexed payload data from several packets within a multiplexed packet (90) to a complete header (91) usable within said network; and
adapting said computing device to process at least one multiplexed packet including:

a payload portion (92a,b,c), formed from a packet transported on said network;
a mini-header (94a,b,c) comprising a channel identifier (98), identifying payload data intended to be exchanged between the same source and the same destination, and a length field (100) indicating the length of said associated payload data portion, the mini-header being usable by said computing device in conjunction with said mapping table (46,48) to construct a packet (90) including said associated payload portion that may be transported on said network;
a mapping token (96) including a channel identifier (102), a payload length field indicating a length of zero (104) and the header of one of said packets (106), the mapping token being usable by said computing device to update said mapping table (46,48), the mapping information indicating a relationship between said mini-header (94a,b,c) and the complete header (91) for transporting said payload data on said network.

18. The memory of claim 17, wherein the channel identifier is 8 bits in length, and the payload data portion (92a,b,c) for each multiplexed packet is less than or equal to 256 bytes in length.

19. The memory of any one of claims 17 to 18, wherein said mini-header (94a,b,c) is shorter than the complete header (91).

20. A method of multiplexing payload data contained in a plurality of packets to be passed by way of a network node in a switched packet network, comprising:

for each packet to be passed by way of said network node, searching a table (46,48) with information from a header of said each packet for a match;
on not finding a match for a given packet having a given header (91), entering information from said given header (91) in said table (46,48) in association with information for a mini-header (94a,b,c) which comprises a channel identifier (98), identifying payload data intended to be exchanged between the same source and the same destination, and a length field (100) indicating the length of said associated payload data

portion;
creating a mapping token (96) associating said given header information (91) with said mini-header (94a,b,c), the mapping token (96) including a channel identifier (102), a payload length field indicating a length of zero (104) and the header of one of said packets (106);
extracting a payload portion (92a,b,c) of said given packet, discarding said given header (91) and associating said mini-header (94a,b,c) with said payload portion (92a,b,c); constructing a multiplexed packet comprising said mapping token (96), said payload portion (92a,b,c) and said associated mini-header (94a,b,c).

21. A computer readable memory storing computer executable instructions, executable by a computing device, to perform the method steps of claim 20.

Patentansprüche

1. Verfahren zur Bildung einer Umsetzungstabelle (46, 48) an einem Netzwerk-Knoten in einem paketvermittelten Netzwerk, wobei die Umsetzungstabelle zur Umsetzung von Mini-Kopffeldern, die multiplexierte Nutzdaten von verschiedenen Paketen innerhalb eines multiplexierten Paketes identifizieren, auf vollständige Kopffelder dient, die zum Transport von Nutzdaten auf dem Netzwerk brauchbar sind, wobei das Verfahren Folgendes umfasst:

Empfangen eines Paketes (90), das Folgendes umfasst:

zumindest einen Nutzdaten-Teil (92a, b, c) von einem der Pakete;
ein Mini-Kopffeld (94a, b, c), das eine Kanal-Identifikation (98), die Nutzdaten identifiziert, die zwischen der gleichen Quelle und dem gleichen Ziel ausgetauscht werden sollen, und ein Längenfeld (100) umfasst, das die Länge des zugehörigen Nutzdaten-Teils anzeigt;
ein Umsetzungs-Token (96), das eine Kanal-Identifikation (102), ein Nutzdaten-Längenfeld, das eine Länge von Null (104) anzeigt, und das Kopffeld eines der Pakete (106) einschließt, wobei das Umsetzungs-Token die Beziehung zwischen dem Mini-Kopffeld (94a, b, c) und einem vollständigen Kopffeld (91) zum Transport des zumindest einen Datenteils auf dem Netzwerk anzeigt;

Bilden eines neuen Eintrages in der Tabelle (46, 48), der einen Teil (70) des Mini-Kopffeldes und Information (76) einschließt, die von dem Umsetzungs-Token abgeleitet ist und die Beziehung zwischen dem Mini-Kopffeld und dem vollständigen Kopffeld (91) anzeigt.

2. Verfahren nach Anspruch 1, bei dem das Mini-Kopffeld eine Kanal-Identifikation (98, 60, 70) umfasst, die einen logischen Kanal innerhalb des multiplexierten Paketes identifiziert.

3. Verfahren nach Anspruch 2, bei dem die Kanal-Identifikation (98, 60, 70) in der Umsetzungstabelle (46, 48) gespeichert wird.

4. Verfahren nach Anspruch 1, bei dem der Netzwerk-Knoten einen Router auf dem paketvermittelten Daten-Netzwerk umfasst.

5. Verfahren nach Anspruch 1, bei dem die Umsetzungstabelle jedesmal dann aktualisiert wird, wenn ein Mini-Kopffeld (94a, b, c), das eine bereits definierte Kanal-Identifikation (98, 60, 70) einschließt, empfangen wird und anzeigt, dass eine dem logischen Kanal zugeordnete Netzwerk-Kommunikation aktiv ist.

6. Verfahren nach Anspruch 5, das weiterhin das Löschen eines Eintrages umfasst, für den ein Mini-Kopffeld (94a, b, c), das eine bereits definierte Kanal-Identifikation einschließt, nicht über ein definiertes Intervall empfangen wird.

7. Verfahren nach Anspruch 1, bei dem die Kanal-Identifikation eine Länge von 8 Bit aufweist.

8. Verfahren nach einem der vorhergehenden Ansprüche, bei dem der Nutzdaten-Teil (92a, b, c) für jedes multiplexierte Paket eine Länge aufweist, die kleiner oder gleich 256 Bytes ist.

9. Verfahren nach einem der vorhergehenden Ansprüche, bei dem das Mini-Kopffeld (94a, b, c) kürzer als das vollständige Kopffeld (91) ist.

10. Verfahren nach Anspruch 9, bei dem das Mini-Kopffeld (94a, b, c) eine Länge von 16 Bits aufweist.

11. Rechenvorrichtung, die mit einem paketvermittelten Daten-Netzwerk verbunden ist, mit einem computerlesbaren Speicher, der Folgendes speichert:

eine Umsetzungstabelle (46, 48) zur Umsetzung von Mini-Kopffeldern, die multiplexierte Nutzdaten von mehreren Paketen innerhalb eines multiplexierten Paketes identifizieren, auf ein vollständiges Kopffeld (91), das in dem paketvermittelten Daten-Netzwerk brauchbar ist;
computerlesbare Befehle, die die Vorrichtung zur Verarbeitung multiplexierter Pakete ausbilden, die:

einen oder mehrere Nutzdaten-Teile (92a, b, c), die aus Paketen auf dem Netzwerk gebildet werden;
ein oder mehrere Mini-Kopffelder (94a, b, c), die eine Kanal-Identifikation (98), die Nutzdaten identifiziert, die zwischen der gleichen Quelle und dem gleichen Ziel ausgetauscht werden sollen, und ein Längenfeld (100) umfasst, das die Länge des zugehörigen Nutzdaten-Teils anzeigt, wobei die Mini-Kopffelder von der Vorrichtung in Verbindung mit der Umsetzungstabelle (46, 48) verwendbar sind, um ein Paket zu konstruieren, das die zugehörigen Nutzdaten einschließt und das auf dem paketvermittelten Daten-Netzwerk transportiert werden kann;
zumindest ein Umsetzungs-Token (96), das eine Kanal-Identifikation (102), ein Nutzdaten-Längenfeld, das eine Länge von Null (104) anzeigt, und das Kopffeld eines der Pakete (106) einschließt, wobei die Umsetzungs-Token von der Vorrichtung zur Aktualisierung der Umsetzungstabelle verwendbar sind, wobei das Umsetzungs-Token (96) eine Beziehung zwischen dem Mini-Kopffeld (94a, b, c) und dem vollständigen Kopffeld (91) zum Transport des Nutzdaten-Teils auf dem Netzwerk anzeigt.

12. Vorrichtung nach Anspruch 11, bei der die computerlesbaren Befehle die Vorrichtung so anpassen, dass sie multiplexierte Pakete demultiplexiert, wobei die Mini-Kopffelder (94a, b, c), die Nutzdaten-Teile (92a, b, c) und die Umsetzungstabellen (46, 48) verwendet werden, um Pakete zu konstruieren, die auf dem Netzwerk transportierbar sind und die Nutzdaten enthalten.

13. Vorrichtung nach Anspruch 11, bei der die Kanal-Identifikation eine Länge von 8 Bit aufweist.

14. Vorrichtung nach einem der Ansprüche 11-13, bei der der Nutzdaten-Teil (92a, b, c) für jedes multiplexierte Paket eine Länge von weniger oder gleich 256 Bytes aufweist.

15. Vorrichtung nach einem der Ansprüche 11-14, bei der das Mini-Kopffeld (94a, b, c) kürzer als das vollständige Kopffeld (91) ist.

16. Vorrichtung nach Anspruch 15, bei dem das Mini-Kopffeld (94a, b, c) eine Länge von 16 Bits aufweist.

17. Computerlesbarer Speicher, der Computer-ausführbare Befehle speichert, die von einer Computereinrichtung ausführbar sind, die mit einem paketvermittelten Netzwerk verbunden ist, wobei die computerlesbaren Befehle die Computereinrichtung so ausbilden, dass:

sie eine Umsetzungstabelle (46, 48) zum Umsetzen von Mini-Kopffeldern (94a, b, c), die multiplexierte Nutzdaten von mehreren Paketen in einem multiplexierten Paket (90) identifizieren, auf ein vollständiges Kopffeld (91) führt, das in dem Netzwerk verwendbar ist; und
sie die Computereinrichtung so anpasst, dass sie das zumindest eine multiplexierte Paket bearbeitet, das Folgendes einschließt:

einen Nutzdaten-Teil (92a, b, c), der aus einem auf dem Netzwerk transportierten Paket gebildet ist;
ein Mini-Kopffeld (94a, b, c), das eine Kanal-Identifikation (98), die Nutzdaten identifiziert, die zwischen der gleichen Quelle und dem gleichen Ziel auszutauschen sind, und ein Längenfeld (100) umfasst, das die Länge des zugehörigen Nutzdaten-Teils anzeigt, wobei das Mini-Kopffeld von der Computereinrichtung in Verbindung mit der Umsetzungstabelle (46, 48) verwendbar ist, um ein Paket (90) zu konstruieren, das den zugehörigen Nutzdaten-Teil einschließt, der auf dem Netzwerk transportiert werden kann;
ein Umsetzungs-Token (96), das eine Kanal-Identifikation (102), ein Nutzdaten-Längenfeld, das eine Länge von Null (104) anzeigt, und das Kopffeld eines der Pakete (106) einschließt, wobei das Umsetzungs-Token von der Computereinrichtung verwendbar ist, um die Umsetzungstabelle (46, 48) zu aktualisieren, wobei die Umsetzungs-Information die Beziehung zwischen dem Mini-Kopffeld (94a, b, c) und dem vollständigen Kopffeld (91) zum Transport der Nutzdaten auf dem Netzwerk anzeigt.

18. Speicher nach Anspruch 17, bei dem die Kanal-Identifikation eine Länge von 8 Bits aufweist, und der Nutzdaten-Teil (92a, b, c) für jedes multiplexierte Paket eine Länge von weniger oder gleich 256 Bytes aufweist.

19. Speicher nach einem der Ansprüche 17-18, bei dem das Mini-Kopffeld (94a, b, c) kürzer als ein vollständiges Kopffeld (91) ist.

20. Verfahren zum Multiplexieren von Nutzdaten, die in einer Vielzahl von Paketen enthalten sind, die über einen Netzwerk-Knoten in einem paketvermittelten Netzwerk weiterzuleiten sind, mit den folgenden Schritten:

für jedes über dem Netzwerk-Knoten weiterzuleitendes Paket, Durchsuchen einer Tabelle (46, 48) mit Information von einem Kopffeld von jedem der Pakete auf eine Übereinstimmung;
wenn keine Übereinstimmung für ein vorgegebenes Paket mit einem vorgegebenen Kopffeld (91) gefunden wird, Eingabe von Information von dem vorgegebenen Kopffeld (91) in die Tabelle (46, 48) in Zuordnung mit Information für ein Mini-Kopffeld (94a, b, c), das eine Kanal-Identifikation (98), die Nutzdaten identifiziert, die zwischen der gleichen Quelle und dem gleichen Ziel ausgetauscht werden sollen, und ein Längenfeld (100) umfasst, das die Länge des zugehörigen Nutzdaten-Teils anzeigt;
Schaffen eines Umsetzungs-Tokens (96), das die vorgegebene Kopffeld-Information (91) zu dem Mini-Kopffeld (94a, b, c) zuordnet, wobei das Umsetzungs-Token (96) eine Kanal-Identifikation (102), ein Nutzdaten-Längenfeld, das eine Länge von Null (104) anzeigt, und das Kopffeld eines der Pakete (106) einschließt;
Extrahieren eines Nutzdaten-Teils (92a, b, c) des vorgegebenen Paketes; Verwerfen des vorgegebenen Kopffeldes (91) und Zuordnen des Mini-Kopffeldes (94a, b, c) zu dem Nutzdaten-Teil (92a, b, c); Konstruieren eines multiplexierten Paketes, das das Umsetzungs-Token (96), den Nutzdaten-Teil (92a, b, c) und das zugehörige Mini-Kopffeld (94a, b, c) umfasst.

21. Computerlesbarer Speicher, der Computer-ausführbare Befehle speichert, die von einer Computereinrichtung ausführbar sind, um die Verfahrensschritte nach Anspruch 20 auszuführen.

Revendications

1. Procédé pour former une table de correspondance (46, 48) au niveau d'un noeud de réseau au sein d'un réseau à commutation de paquets, ladite table de correspondance étant prévue pour faire correspondre des mini en-têtes identifiant des données de charge utile multiplexées issues de plusieurs paquets au sein d'un paquet multiplexé avec des en-têtes complets utilisables pour transporter de la charge utile sur ledit réseau, ledit procédé comprenant les étapes consistant à :

recevoir un paquet (90) comprenant :

au moins une partie de données de charge utile (92a, b, c) d'un desdits paquets ;
un mini en-tête (94a, b, c) comprenant un identificateur de canal (98) identifiant des données de charge utile à échanger entre la même source et la même destination, et un champ de longueur (100) indiquant la longueur de ladite partie de données de charge utile associée ;
un jeton de mise en correspondance (96) comprenant un identificateur de canal (102), un champ de longueur de charge utile indiquant une longueur de zéro (104) et l'en-tête d'un desdits paquets (106), le jeton de mise en correspondance étant indicatif d'une relation entre ledit mini en-tête (94a, b, c) et un en-tête complet (91) pour transporter ladite au moins une partie de données sur ledit réseau ;

former une nouvelle entrée au sein de ladite table (46, 48), comprenant une partie (70) dudit mini en-tête et des informations (76) dérivées dudit jeton de mise en correspondance, indicatives d'une relation entre ledit mini en-tête et ledit en-tête complet (91).

2. Procédé selon la revendication 1, dans lequel ledit mini en-tête comprend un identificateur de canal (98, 60, 70), identifiant un canal logique au sein dudit paquet multiplexé.

3. Procédé selon la revendication 2, dans lequel ledit identificateur de canal (98, 60, 70) est stocké au sein de ladite table de correspondance (46, 48).

4. Procédé selon la revendication 1, dans lequel ledit noeud de réseau comprend un routeur sur ledit réseau de données à commutation de paquets.

5. Procédé selon la revendication 1, dans lequel ladite table de correspondance est mise à jour à chaque fois qu'un mini en-tête (94a, b, c) comprenant un identificateur de canal déjà défini (98, 60, 70) est reçu, indiquant qu'une communication de réseau associée au dit canal logique est active.

6. Procédé selon la revendication 5, comprenant en outre l'étape consistant à supprimer une entrée pour laquelle un mini en-tête (94a, b, c) comprenant un identificateur de canal déjà défini n'est pas reçu pendant un intervalle défini.

7. Procédé selon la revendication 1, dans lequel l'identificateur de canal a une longueur de 8 bits.

8. Procédé selon l'une quelconque des revendications précédentes, dans lequel la partie de données de charge utile (92a, b, c) pour chaque paquet multiplexé a une longueur inférieure ou égale à 256 octets.

9. Procédé selon l'une quelconque des revendications précédentes, dans lequel ledit mini en-tête (94a, b,

c) est plus court que l'en-tête complet (91).

10. Procédé selon la revendication 9, dans lequel le mini en-tête (94a, b, c) a une longueur de 16 bits.

11. Appareil informatique interconnecté à un réseau de données à commutation de paquets comprenant une mémoire lisible par ordinateur stockant :

une table de correspondance (46, 48) pour faire correspondre des mini en-têtes identifiant des données de charge utile multiplexées issues de plusieurs paquets au sein d'un paquet multiplexé avec un en-tête complet (91) utilisable au sein dudit réseau de données à commutation de paquets ;
des instructions lisibles par ordinateur, adaptant ledit appareil pour traiter des paquets multiplexés comprenant :

une ou plusieurs parties de charge utile (92a, b, c) formées à partir de paquets sur ledit réseau ;
un ou plusieurs mini en-têtes (94a, b, c) comprenant un identificateur de canal (98) identifiant des données de charge utile à échanger entre la même source et la même destination, et un champ de longueur (100) indiquant la longueur de ladite partie de données de charge utile associée, les mini en-têtes étant utilisables par ledit appareil en conjonction avec ladite table de correspondance (46, 48) pour construire un paquet comprenant ladite charge utile associée qui peut être transportée sur ledit réseau de données à commutation de paquets ;
au moins un jeton de mise en correspondance (96) comprenant un identificateur de canal (102), un champ de longueur de charge utile indiquant une longueur de zéro (104) et l'en-tête d'un desdits paquets (106), les jetons de mise en correspondance étant utilisables par ledit appareil pour mettre à jour ladite table de correspondance, le jeton de mise en correspondance (96) étant indicatif d'une relation entre ledit mini en-tête (94a, b, c) et l'en-tête complet (91) pour transporter ladite partie de données sur ledit réseau.

12. Appareil selon la revendication 11, dans lequel lesdites instructions lisibles par ordinateur adaptent ledit appareil pour démultiplexer des paquets multiplexés en utilisant lesdits mini en-têtes (94a, b, c), lesdites parties de charge utile (92a, b, c) et lesdites tables de correspondance (46, 48) pour construire des paquets transportables sur ledit réseau et contenant lesdites données de charge utile.

13. Appareil selon la revendication 11, dans lequel l'identificateur de canal a une longueur de 8 bits.

14. Appareil selon l'une quelconque des revendications 11 à 13, dans lequel la partie de données de charge utile (92a, b, c) pour chaque paquet multiplexé a une longueur inférieure ou égale à 256 octets.

15. Appareil selon l'une quelconque des revendications 11 à 14, dans lequel ledit mini en-tête (94a, b, c) est plus court que l'en-tête complet (91).

16. Procédé selon la revendication 15, dans lequel le mini en-tête (94a, b, c) a une longueur de 16 bits.

17. Mémoire lisible par ordinateur stockant des instructions exécutables par un ordinateur, exécutables par un dispositif informatique interconnecté à un réseau à commutation de paquets, lesdites instructions lisibles par ordinateur adaptant ledit dispositif informatique pour :

maintenir une table de correspondance (46, 48) pour faire correspondre des mini en-têtes (94a, b, c) identifiant des données de charge utile multiplexées issues de plusieurs paquets au sein d'un paquet multiplexé (90) avec un en-tête complet (91) utilisable au sein dudit réseau de données ; et
adapter ledit dispositif informatique pour traiter au moins un paquet multiplexé comprenant :

une partie de charge utile (92a,b c) formée à partir d'un paquet transporté sur ledit réseau ;
un mini en-tête (94a, b, c) comprenant un identificateur de canal (98) identifiant des données de charge utile à échanger entre la même source et la même destination, et un champ de longueur (100) indiquant la longueur de ladite partie de données de charge utile associée, le mini en-tête étant utilisable par ledit dispositif informatique en conjonction avec ladite table de correspondance (46, 48) pour construire un paquet (90) comprenant ladite partie de charge utile associée qui peut être transportée sur ledit réseau ;
un jeton de mise en correspondance (96) comprenant un identificateur de canal (102), un champ de longueur de charge utile indiquant une longueur de zéro (104) et l'en-tête d'un desdits paquets (106), le jeton de mise en correspondance étant utilisable par ledit dispositif informatique pour mettre à jour ladite table de correspondance (46,

48), les informations de mise en correspondance indiquant une relation entre ledit mini en-tête (94a, b, c) et l'en-tête complet (91) pour transporter lesdites données de charge utile sur ledit réseau.

18. Mémoire selon la revendication 17, dans laquelle l'identificateur de canal a une longueur de 8 bits, et la partie de données de charge utile (92a, b, c) pour chaque paquet multiplexé a une longueur inférieure ou égale à 256 octets.

19. Mémoire selon l'une quelconque des revendications 17 à 18, dans laquelle ledit mini en-tête (94a, b, c) est plus court que l'en-tête complet (91).

20. Procédé de multiplexage de données de charge utile contenues dans une pluralité de paquets à transmettre par l'intermédiaire d'un noeud de réseau dans un réseau à commutation de paquets comprenant les étapes consistant à :

pour chaque paquet à transmettre par l'intermédiaire dudit noeud de réseau, rechercher une correspondance dans une table (46, 48) à l'aide d'informations provenant d'un en-tête dudit chaque paquet ;
lorsqu'aucune correspondance n'est trouvée pour un paquet donné ayant un en-tête donné (91), entrer des informations à partir dudit en-tête donné (91) dans ladite table (46, 48) en association avec des informations pour un mini en-tête (94a, b, c) qui comprend un identificateur de canal (98), identifiant des données de charge utile destinées à être échangées entre la même source et la même destination, et un champ de longueur (100) indiquant la longueur de ladite partie de données de charge utile associée;
créer un jeton de mise en correspondance (96) associant lesdites informations de l'en-tête donné (91) avec ledit mini en-tête (94a, b, c), le jeton de mise en correspondance (96) comprenant un identificateur de canal (102), un champ de longueur de charge utile indiquant une longueur de zéro (104) et l'en-tête d'un desdits paquets (106) ;
extraire une partie de charge utile (92a, b, c) dudit paquet donné, supprimer ledit en-tête donné (91) et associer ledit mini en-tête (94a, b, c) avec ladite partie de charge utile (92a, b, c) ;
construire un paquet multiplexé comprenant ledit jeton de mise en correspondance (96), ladite partie de charge utile (92a, b, c) et ledit mini en-tête associé (94a, b, c).

21. Mémoire lisible par ordinateur stockant des instructions exécutables par ordinateur, exécutables par un dispositif informatique pour exécuter les étapes du procédé de la revendication 20.

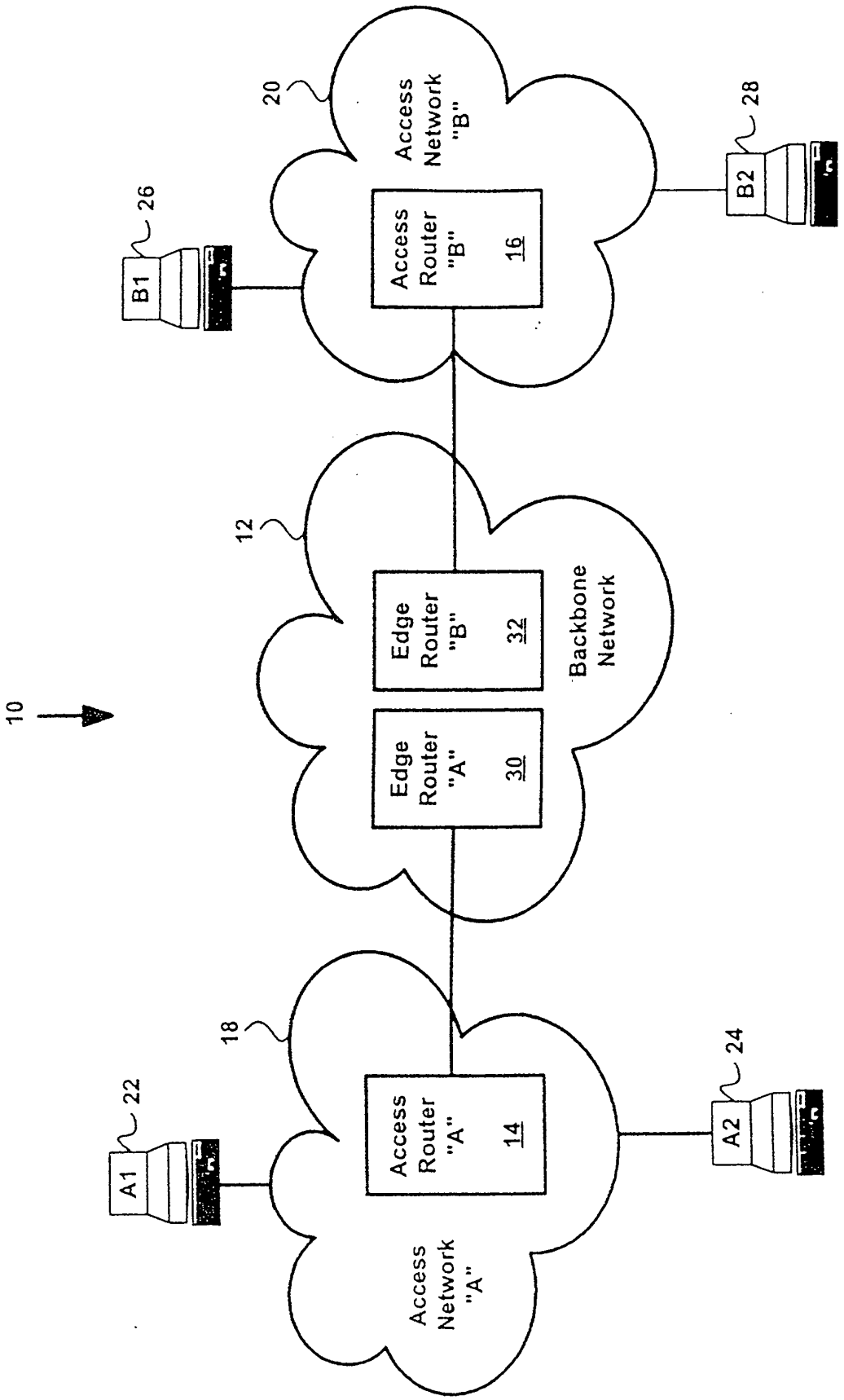
10
12
Backbone Network
Edge Router "B" 32
Edge Router "A" 30
18
Access Network "A"
Access Router "A" 14
A1
22
A2
24
20
Access Network "B"
Access Router "B" 16
B1
26
B2
28

FIG. 1

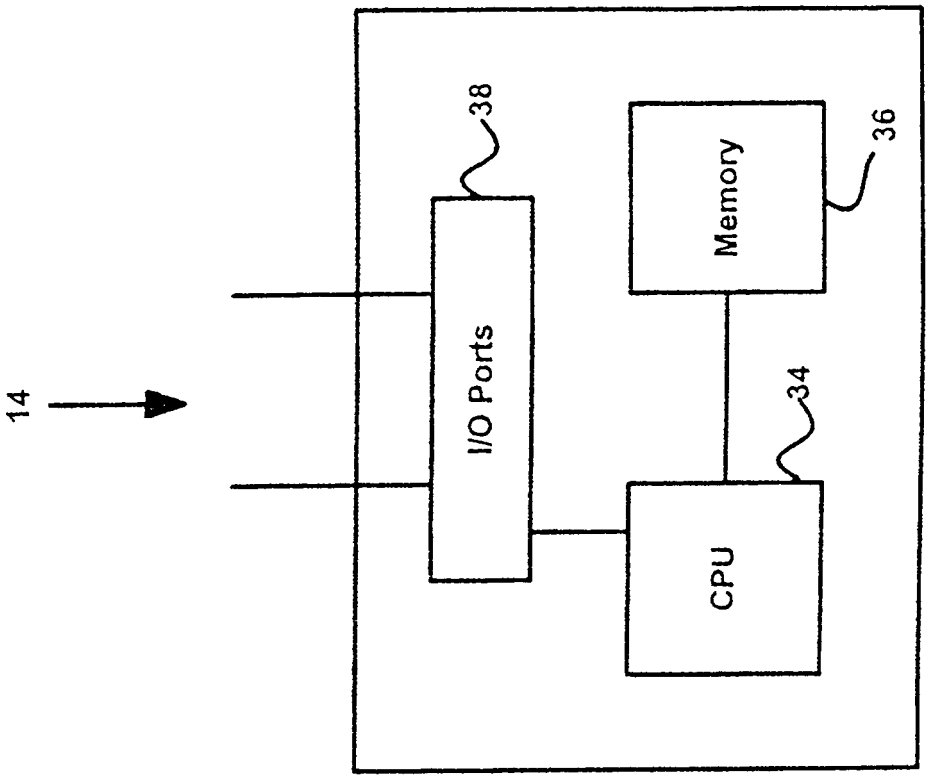

FIG. 2

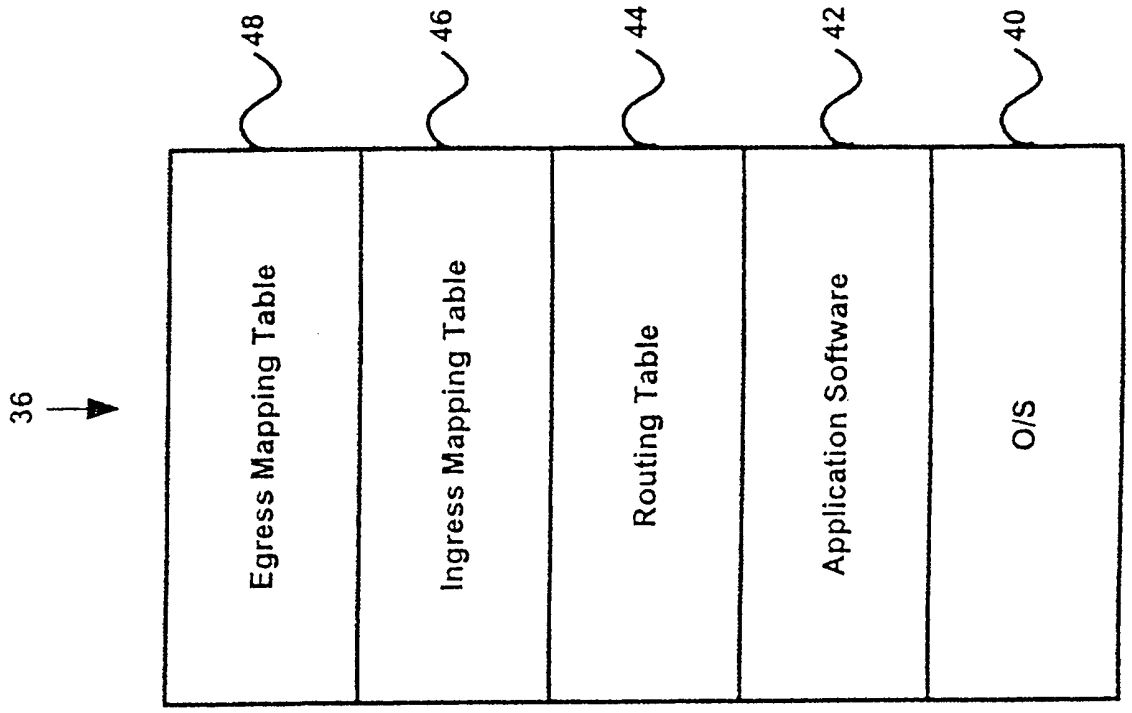
36
48
46
44
42
40
Egress Mapping Table
Ingress Mapping Table
Routing Table
Application Software
O/S

FIG. 3

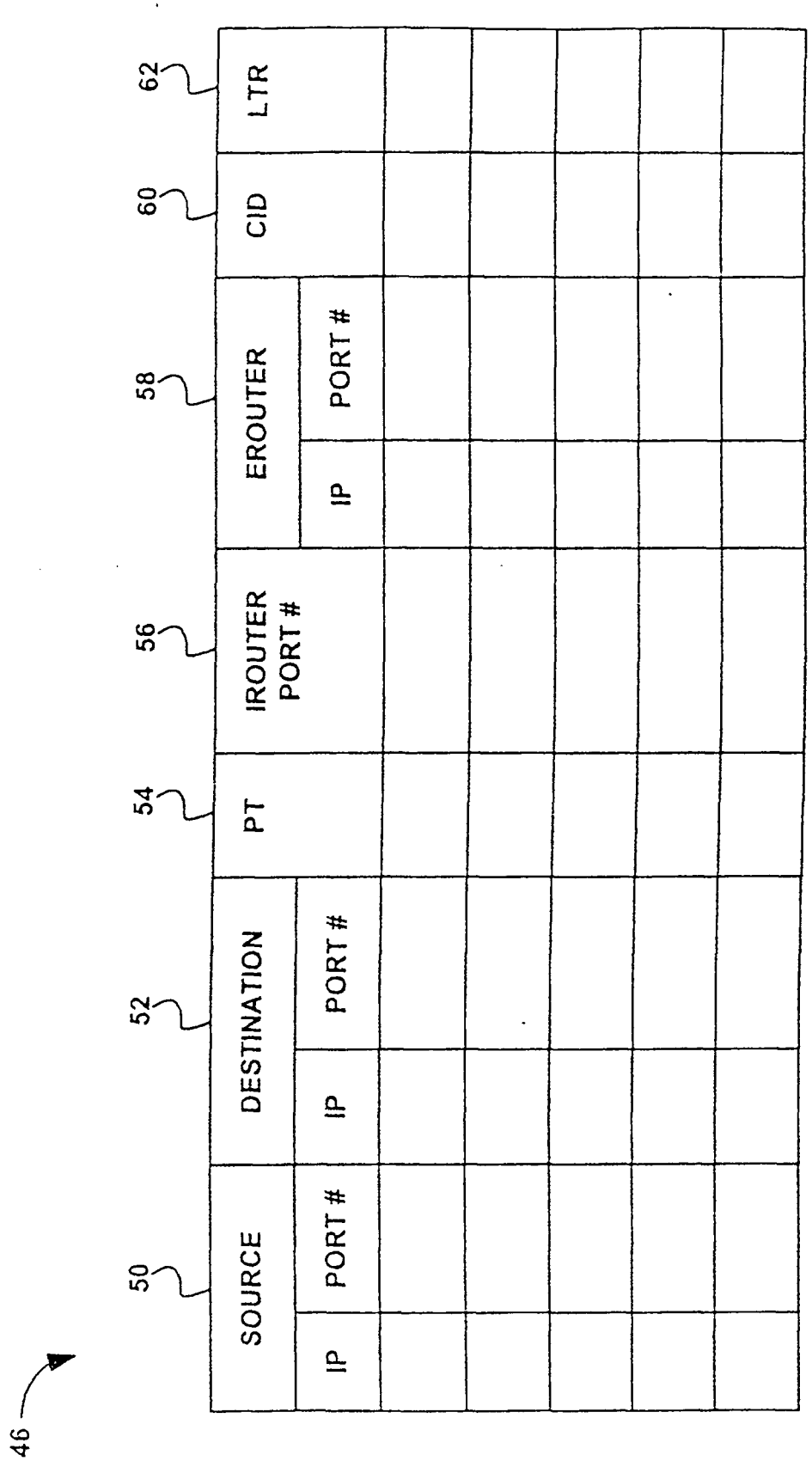
46
50
SOURCE
IP
PORT #
52
DESTINATION
IP
PORT #
54
PT
56
IROUTER PORT #
58
EROUTER
IP
PORT #
60
CID
62
LTR

FIG. 4

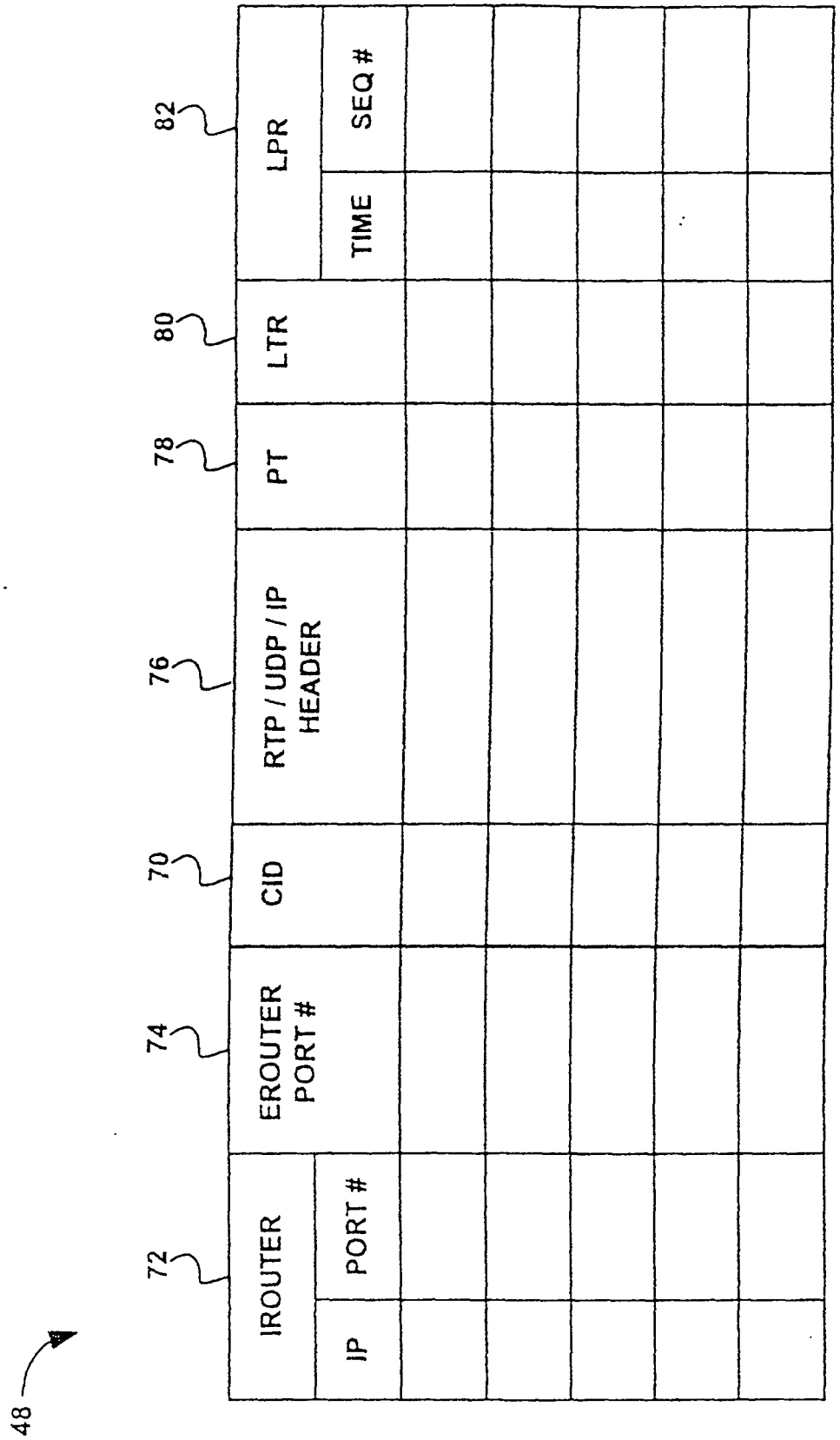
48
72
IROUTER
IP
PORT #
74
EROUTER PORT #
70
CID
76
RTP / UDP / IP HEADER
78
PT
80
LTR
82
LPR
TIME
SEQ #

FIG. 5

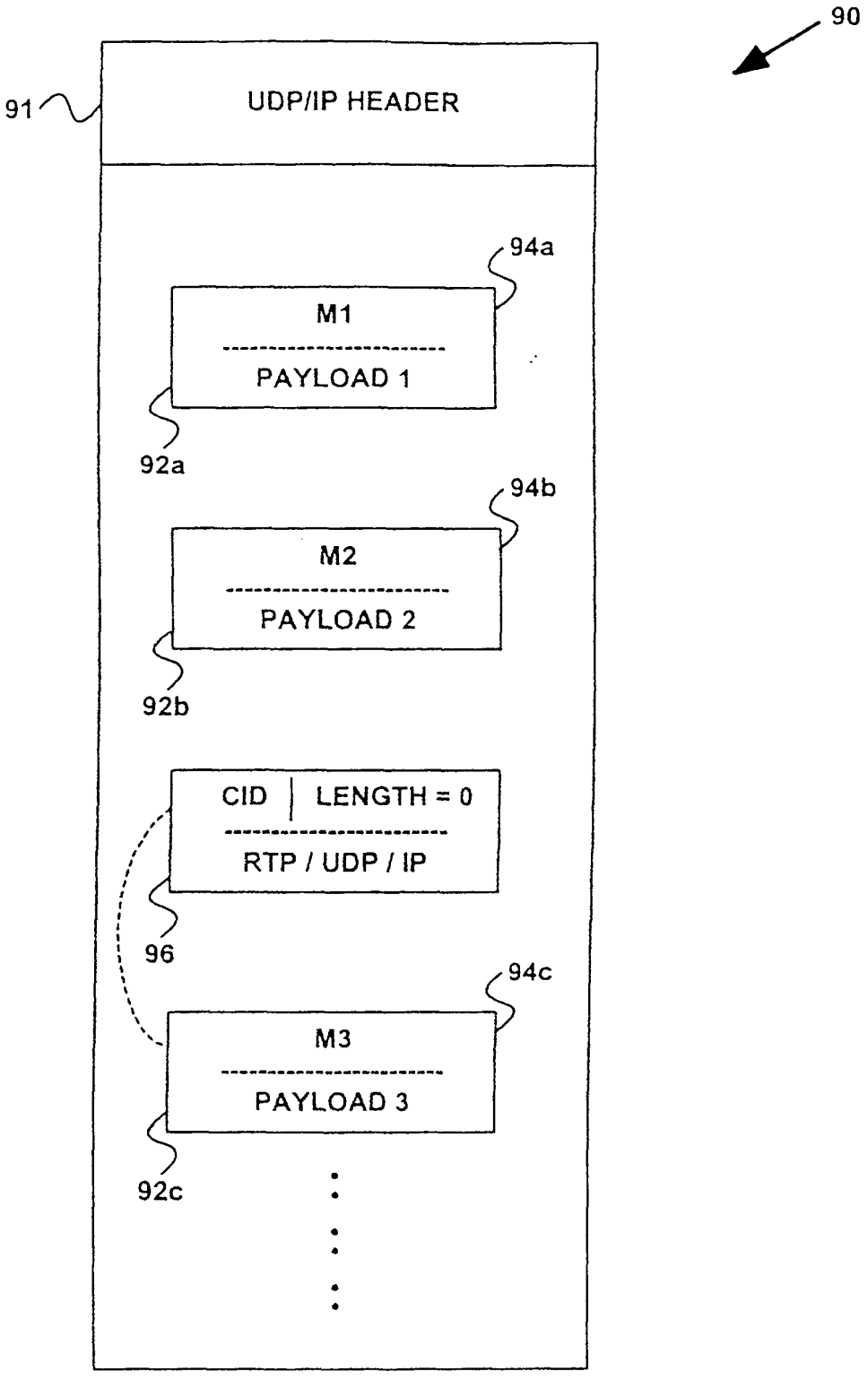
90
91
UDP/IP HEADER
94a
M1
PAYLOAD 1
92a
94b
M2
PAYLOAD 2
92b
CID | LENGTH = 0
RTP / UDP / IP
96
94c
M3
PAYLOAD 3
92c

FIG. 6

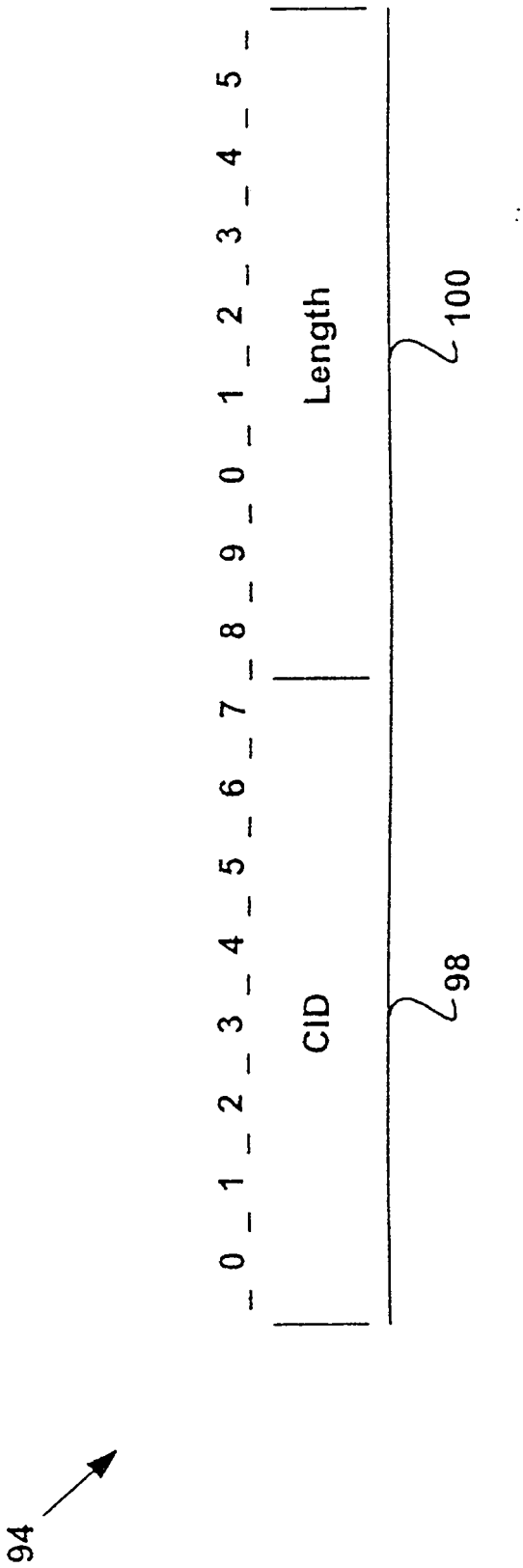

FIG. 7

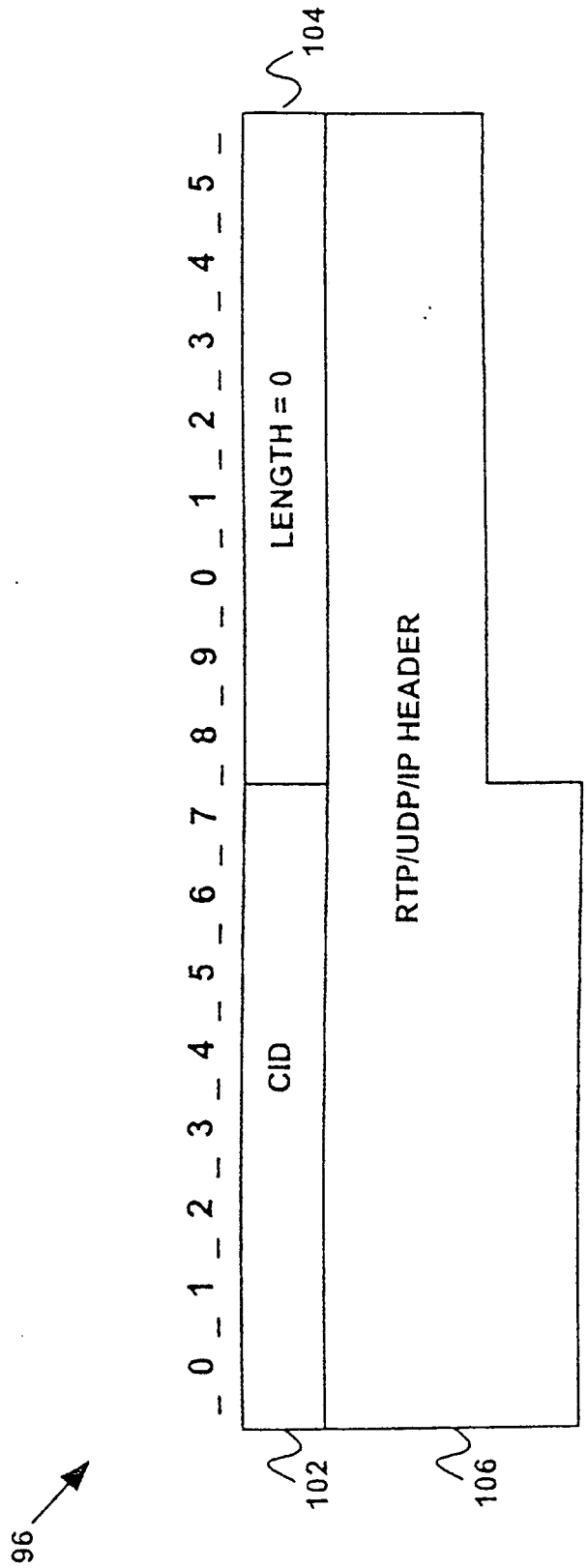
96
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
CID
LENGTH = 0
RTP/UDP/IP HEADER
102
104
106

FIG. 8

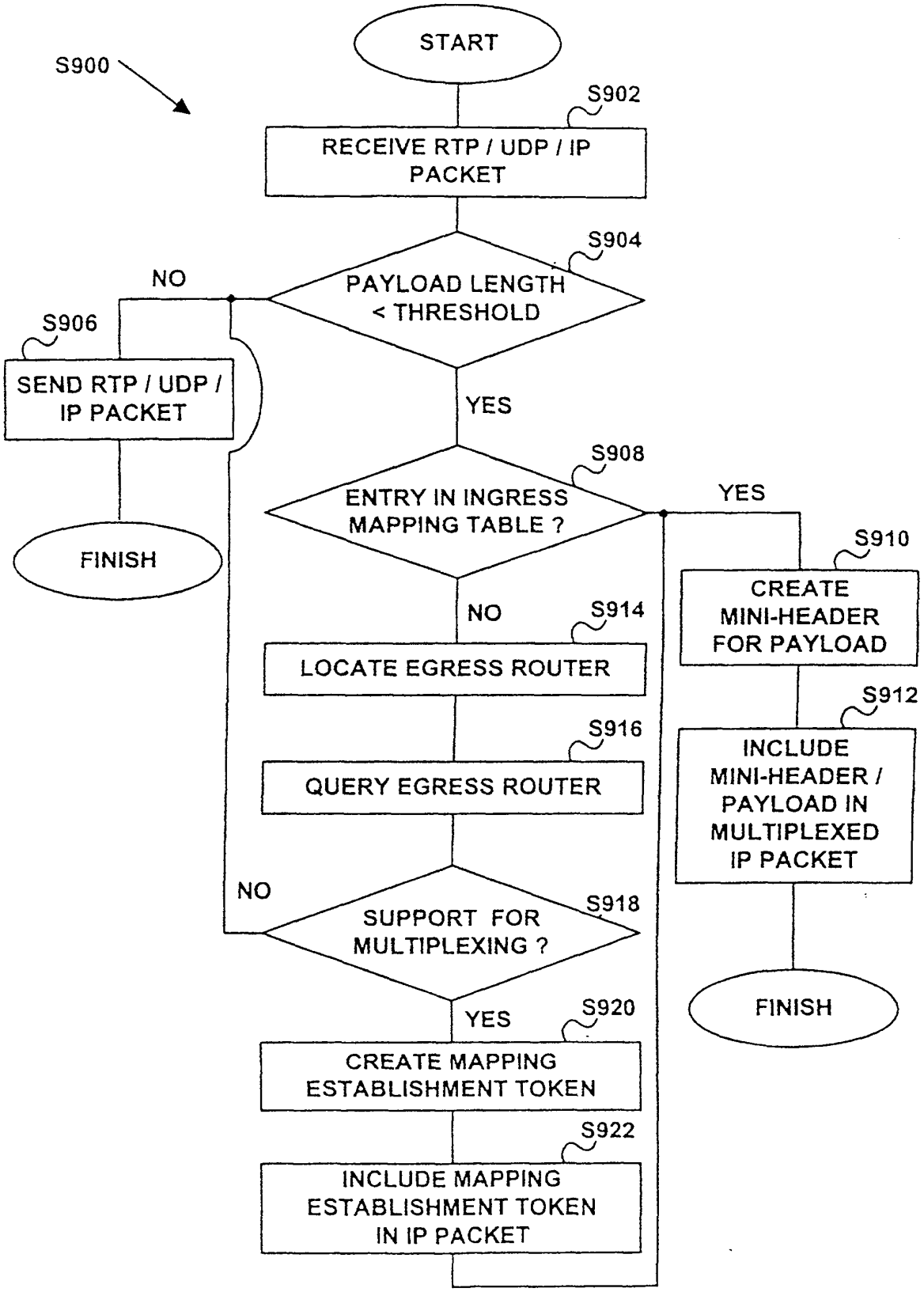
S900
START
S902
RECEIVE RTP / UDP / IP PACKET
S904
PAYLOAD LENGTH < THRESHOLD
NO
S906
SEND RTP / UDP / IP PACKET
FINISH
YES
S908
ENTRY IN INGRESS MAPPING TABLE ?
YES
S910
CREATE MINI-HEADER FOR PAYLOAD
S912
INCLUDE MINI-HEADER / PAYLOAD IN MULTIPLEXED IP PACKET
FINISH
NO
S914
LOCATE EGRESS ROUTER
S916
QUERY EGRESS ROUTER
NO
S918
SUPPORT FOR MULTIPLEXING ?
YES
S920
CREATE MAPPING ESTABLISHMENT TOKEN
S922
INCLUDE MAPPING ESTABLISHMENT TOKEN IN IP PACKET

FIG. 9

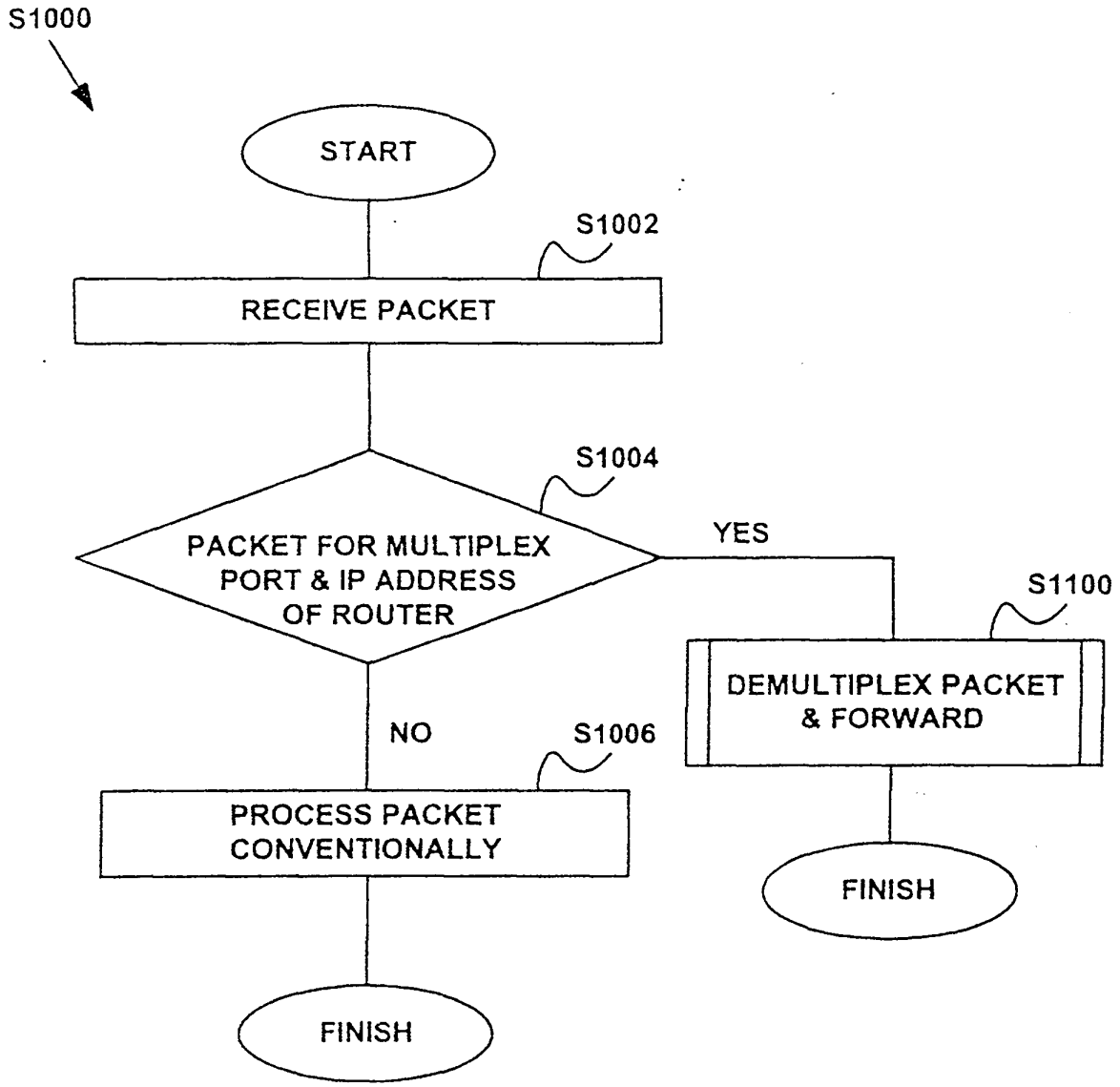
S1000
START
S1002
RECEIVE PACKET
S1004
PACKET FOR MULTIPLEX PORT & IP ADDRESS OF ROUTER
YES
S1100
DEMULTIPLEX PACKET & FORWARD
NO
S1006
PROCESS PACKET CONVENTIONALLY
FINISH
FINISH

FIG. 10

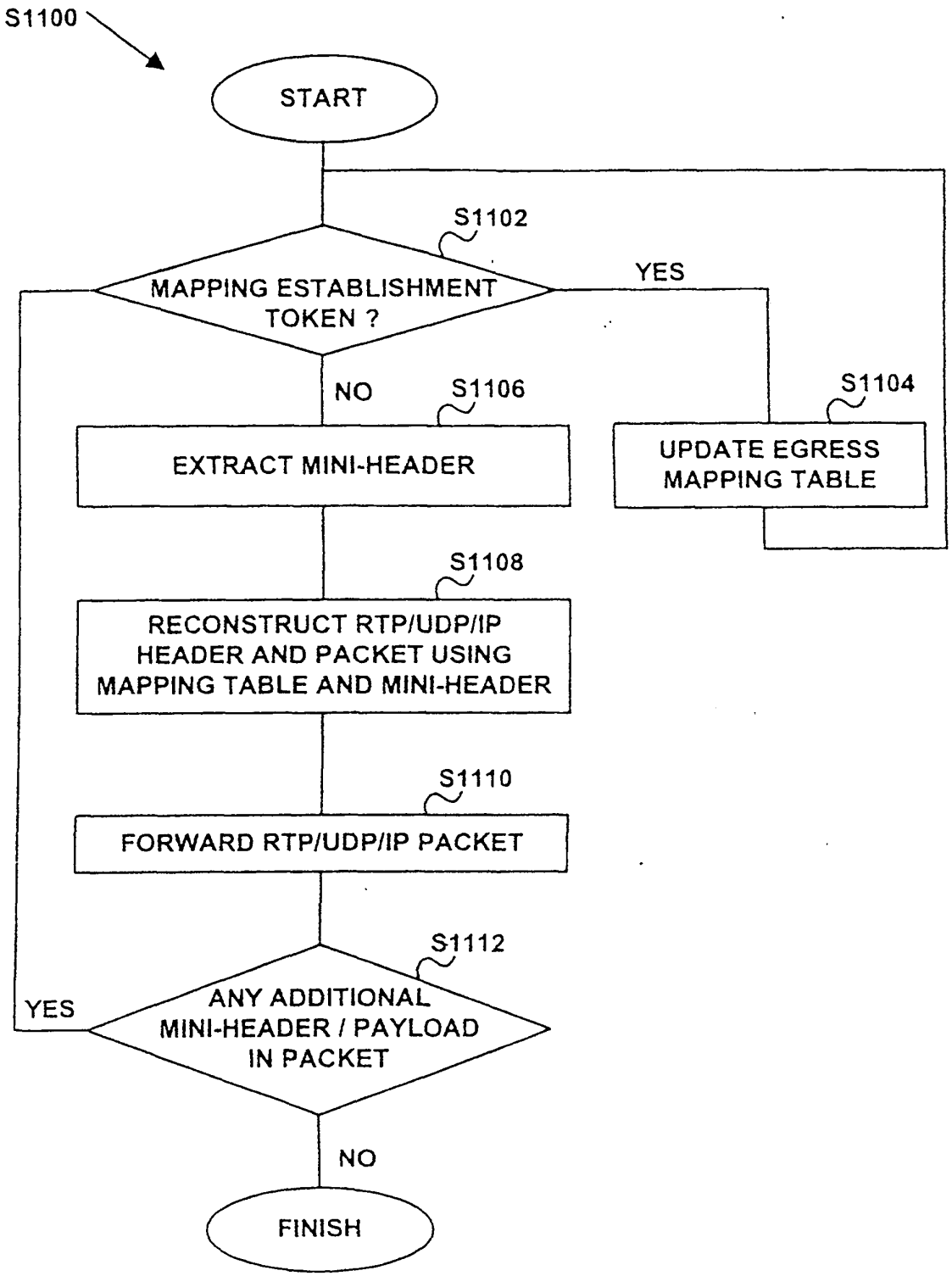
S1100
START
S1102
MAPPING ESTABLISHMENT TOKEN ?
YES
NO
S1106
EXTRACT MINI-HEADER
S1104
UPDATE EGRESS MAPPING TABLE
S1108
RECONSTRUCT RTP/UDP/IP HEADER AND PACKET USING MAPPING TABLE AND MINI-HEADER
S1110
FORWARD RTP/UDP/IP PACKET
S1112
ANY ADDITIONAL MINI-HEADER / PAYLOAD IN PACKET
YES
NO
FINISH

FIG. 11